

Zefeng (Daniel) Wang

Availability: Full time starting January 2022

7 Wycombe Way, Princeton Junction, NJ
(609) 454-1717 | dnl.wang@gmail.com | zefeng-wang.com
linkedin.com/in/zefeng-daniel-wang | github.com/zefwang

EDUCATION

Northeastern University, Boston, MA

December 2021

Khoury College of Computer Sciences

GPA: 3.78/4.00

Candidate for Bachelor of Science in Computer Science & Business Administration (Finance)

Minor in Mathematics

Relevant Courses: Algorithms and Data, Object Oriented Design, Linear Algebra, Computer Systems, Database Design

Honors and Awards: Dean's List, International Scholar, Top 10/Best Rookie Award at Hack Beanpot 2019

COMPUTER KNOWLEDGE

Languages: Java, JavaScript/TypeScript (NodeJS, React), HTML, CSS, Golang, Python, SQL, C,

Tools/Libraries: Docker, Firebase, Google Cloud (Firestore, Functions, Storage), JUnit, Kubernetes, Unix, NumPy, pandas

Miscellaneous: Git, GitHub, JSON, npm, RESTful APIs, XML, YAML

EXPERIENCE

Software Developer Co-op @ Intuit

May 2020 – December 2020

- Improved accessibility of metrics using Java and Go on the Observability team for QuickBooks Online product
- Developed API in Go by implementing schema & CRUD functions on a MySQL database, configuring Elasticsearch index to improve search efficiency, and setting up infrastructure for Kubernetes cluster
- Used Micrometer and Hystrix (Java Spring) to create a filter for unnecessary data to improve readability of metrics
- Adopted Spring's aspect-oriented programming to annotate metrics and decrease code complexity via abstraction

Backend Web Engineer @ BusRight, Inc

November 2019 – November 2020

- Improved web API by adding new features, enforcing test-driven development, and increasing efficiency
- Developed password reset process by adding functionality to get info from database & send emails via SendGrid
- Enforced persistent notifications by writing endpoints to get/send notifications to collections in Cloud Firestore
- Added Cloud Functions for daily database backup and profile picture functions triggered on user deletion/creation

Full-Stack Developer @ Sandbox at Northeastern

September 2019 - Present

- Building mobile application for research client to study memory in older populations using React Native and Expo
- Implemented changes to SearchNEU, to ensure accurate, durable, and accessible data for over 50,000 users

Core Team – Sponsorship @ HackBeanpot Inc

May 2019 – Present

- Engaged in agile environment to organize annual hackathon for approximately 200 Boston students
- Devised sponsorship packet and worked with dozens of companies to raise over \$20,000 in financial backing
- Led team of three in revamping judging system for students' projects to be more holistic and efficient
- Designed algorithm for judging scheduling to resolve shortcomings with prior technology
- Implemented interactive packet on website using React functional components and Gatsby.js

PROJECTS

SearchNEU (NodeJS, React, Typescript)

September 2019 – August 2020

- Moved data to relational database (PostgreSQL) by creating data models using Sequelize ORM
- Developed toggle and dropdown components in React to improve loading times ~30x and enable filtered search
- Rewrote scraper classes that produce info from colleges' sites with Cheerio implementation of jQuery
- Refactored JavaScript backend and enforcing reliability using Jest framework and snapshot testing

Emojify (Google Cloud API, Twilio API, Node.JS)

November 2019

- Built program that translates text message from English into emojis using Twilio's Programmable SMS
- Trained dataset that classifies content using custom categories with Google's AutoML Natural Language API
- Developed using Node to integrate Twilio API and send responses based on result of Python script

CrimePot (Python, Flask)

February 2019

- Implemented Google Maps API with HTML/CSS/JavaScript to place markers representing crimes on map
- Performed HTTP requests using Flask microframework to retrieve data from CKAN action API
- Filtered through JSON data based on user parameters and then converted to GeoJSON in Python